# Adopting Trust and Assurance as indicators for the reassignment of responsibilities in Multi-agent Systems

Gateau, Benjamin; Ouedraogo, Moussa; Feltus, Christophe; Guemkam, Guy; Danoy, Gregoire; Seredinsky, Marcin; U. Khan, Samee; Khadraoui, Djamel; Bouvry, Pascal

*Published in:*
The Knowledge Engineering Review

*DOI:*
http://dx.doi.org/10.1017/S0269888914000290

*Publication date:*
2012

*Document Version*
Early version, also known as pre-print

*The Knowledge Engineering Review*, Vol. 00:0, 1–24. © 2012, Cambridge University Press
DOI: 10.1017/S000000000000000 Printed in the United Kingdom

# Adopting Trust and Assurance as Indicators for the Reassignment of Responsibilities in Multi-Agent Systems

BENJAMIN GÂTEAU[1], MOUSSA OUEDRAOGO[1], CHRISTOPHE FELTUS[1], GUY GUEMKAM[1], GRÉGOIRE DANOY[2], MARCIN SEREDYNSKI[3], SAMEE U. KHAN[4], DJAMEL KHADRAOUI[1] and PASCAL BOUVRY[2]

[1] *Public Research Centre Henri Tudor, 29 Av. John F. Kennedy, L-1855 Luxembourg*
*E-mail: surname.name@tudor.lu*
[2] *CSC Research Unit, University of Luxembourg, 6 rue Coudenhove Kalergi, L-1359 Luxembourg*
*E-mail: {gregoire.danoy,pascal.bouvry}@uni.lu*
[3] *Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg, 6 rue Coudenhove Kalergi, L-1359 Luxembourg*
*E-mail: marcin.seredynski@uni.lu*
[4] *NDSU-CIIT Green Computing and Communications Laboratory, Department of Electrical and Computer Engineering, North Dakota State University, Fargo, ND 58108–6050, USA*
*E-mail: samee.khan@ndsu.edu*

**Abstract**

Multi-agent systems have been widely used in the literature, including for the monitoring of distributed systems. However, one of the unresolved issues in this technology remains the reassignment of the responsibilities of the monitoring agents when some of them become unable to meet their obligations. This paper proposes a new approach for solving the problem based on: (a) the gathering of evidence on whether the agent can or cannot fulfil the tasks it has been assigned and (b) the reassignment of the task to alternative agents using their trust level as a selection parameter. A weather station use case scenario is proposed as an instantiation of the proposed model.

## 1 Introduction

The adoption of multi-agent systems (MAS) to monitor highly distributed systems has been gaining momentum in the recent years. Such systems have witnessed crucial demand for deployment in diverse application scenarios, such as E-commerce, E-health, network intrusion detection, telematics and transport systems, environmental monitoring (Baig, 2012). The rationale for the aforementioned mainly lies on the inherent properties and characteristics that the multi-agent technology offers. An agent is commonly considered as an encapsulated computer system that is situated in some environment and is capable of flexible and autonomous action within the environment to meet the design objectives (Wooldridge, 2002). As agents have control over their own behaviour, they may (and in many cases must) cooperate and negotiate with each other to achieve the desired goals (Jennings, 1999). The convergence of these agents' properties and distributed systems behaviour makes the multi-agent architecture an appropriate mechanism for the monitoring of network infrastructure including the security aspects (Wooldridge, 2002). However, for such a MAS to provide an efficient monitoring of the network, it is imperative that each involved agent meets the assigned objective functions that we will hereafter, refer to as responsibilities. The environment in which the system or network operates can be subject to changes that may not have been foreseen. Therefore, some agents may become unable to meet their assigned responsibilities. For instance, hazards or even malicious actions could lead to

communication links between the agents part of the monitoring system to break. Alternatively, the agents entrusted with conducting the verifications and measurements may fail to fulfil their responsibilities for others range of raisons including: (a) erroneous assignment of their rights or alteration of the latter during runtime, (b) the agents' capabilities may become insufficient for accomplishing whole the tasks assigned to them, and (c) an accumulation of tasks for an agent may result in an overload and subsequently a failure to meet some of them. Consequently, a solution where multi-agents systems are reinforced with the ability to reassign the responsibilities of faulty agents to others that harbour similar capabilities is needed. A prerequisite for the above mentioned is to first gather the body of evidence that could inform on the aptitude or inaptitude of a given agent to meet the corresponding responsibilities. Such assuring information can then form the basis for a trusted decision as to whether alternative agents should be sought for the fulfilment of a given task. Unfortunately, the survey of the relevant literature has revealed that although numerous contributions have been made towards the provision of MAS dedicated to the monitoring of networks and systems — example of (Kolaczek and Juszczyszyn, 2007; Abielmona et al., 2011) — the assignment of the agents' functions is undertaken prior to the MAS system deployment. Moreover, reassignment of some of the responsibilities to others agents in the event when certain agents become unable to carry out the corresponding tasks is impossible. Therefore, a faulty agent may result in the monitoring system being grounded or induce the system to issue monitoring values that are either erroneous or incomplete. Needless to state that the gravity of the consequences of such a failure is stringently related to the criticality of the infrastructure or the system being monitored. Consequently, the core question addressed in this paper can be stated as: "How to ascertain the continuity of the MAS monitoring activities in a way that that the failure of some of the agents to fulfil the corresponding responsibilities does not drastically affect system operations?" The approach presented in this paper relies on the evaluation of the assurance that a given agent has the necessary credentials to fulfil the corresponding responsibility. A decision to automatically reassign to a different agent in case of the occurrence of a hazard that results in one agent being isolated or unable to react promptly on a monitoring request is made based on a trust value. For that, we use a normative organisation modelling language for MAS named $\mathcal{M}$OISE$^{Inst}$ (Gâteau et al., 2005), which is an an extension of the $\mathcal{M}$OISE$^{+}$ developed by (Hübner et al., 2002) to represent security policies, provided by the responsibility model of (Feltus, 2010), as norms and to supervise agent respect. Information linked to the specification of norms is used as metrics to assure achievement. Respect of norms histories are used as input for the centralised evaluation of agent's reputation. $\mathcal{S}$YNAI (Boissier and Gâteau, 2007) is a multi-agent organisation infrastructure that interprets normative declarative organisations programmed with $\mathcal{M}$OISE$^{Inst}$. The system is composed of generic *supervisor* agents, aiming at controlling and enforcing the rights and duties of autonomous "domain" agents operating in an normative organisation expressed with $\mathcal{M}$OISE$^{Inst}$ (Boissier and Gâteau, 2007). *Supervisor* agents will be able, regarding the respect assurance of a norm and the reputation of agents, to re-assign dynamically agents to roles (or responsibilities).

The remainder of this article is organised as follows. Section 2 discusses the related work in terms of organisation modelling languages, responsibility models in MAS, and monitoring of security. Section 3 presents the proposed responsibility model and the condition under which an agent is expected to meet the corresponding responsibility. In Section 4, we describe the reference scenario adopted for the validation of our model. Section 5 presents an instantiation of the responsibility model through the assignment and the reassignment of the responsibilities. Finally, Section 6 provides our conclusions and perspectives.

## 2 Related Work

### *2.1 Organisation modelling languages*

Different models are manipulated within a MAS that can be described by using vowel approach AEIO (Agent, Environment, Interaction, organisation) introduced in (Demazeau, 1995). *Agents*

deal with the models (or architectures) used for the active part of the agent, from a simple automata to a complex knowledge based system. *Environments* are the places where the agents are located. *Interactions* concern the infrastructures, the languages, and the interaction protocols between agents, from simple physical interactions to complex communicative acts. *Organisations* structure the agents in groups, hierarchies and relations.

Our primary focus will be on the latter. That is to say that, how MAS organise agents and set up a self-governing behaviour through cooperation between agents? There exists two possible ways of obtaining an organisation in a MAS, namely: a bottom-up or a top-down approach. Contrary to the top-down approach, the bottom-up one does not manipulate any organisational model defined *a priori*. However, the approach utilises some interaction capabilities to dynamically create and adapt the MAS organisations. Therefore, in this work, we consider only the top-down approaches and detail in the subsequent text the organisation modelling languages that exist in the multi-agent domain. To represent the complex social organisation within a MAS, different modelling dimensions are used, such as structural, functional, dialogic, environmental, and contextual.

The *structural* dimension represents the structure of the collective level of a MAS generally in terms of roles/groups/links. Such structural specification is used in AGR (Ferber and Gutknecht, 1998), ISLANDER (Esteva et al., 2002), $\mathcal{M}$OISE$^{+}$and $\mathcal{M}$OISE$^{Inst}$. The *functional* dimension specifies the global functioning of the system, as used in TAEMS (Lesser et al., 2004), TEAMCORE (Pynadath and Tambe, 2003), or $\mathcal{M}$OISE$^{+}$. Some models, such as ISLANDER, add a *dialogical* dimension that specifies MAS interactions in terms of communications between agents. The *environmental* dimension allows to constrain the anchoring of the organisation in an environment, such as in AGRE (Ferber et al., 2004). Inspired by ISLANDER, $\mathcal{M}$OISE$^{Inst}$ introduced a *contextual* specification to define *a priori* the transition between different configurations of norms, structures, and plans (Boissier and Gâteau, 2007). We do not intend to provide an exhaustive comparison of the aforementioned organisation modelling languages (OML) in terms of the primitives or modelling power that each may can offer. The interested readers are encouraged to refer to (Coutinho et al., 2007) for a systematic comparison of OML.

As mentioned in (Boissier and Gâteau, 2007), depending on the various dimensions, the influence on the agents' behaviour may be quite different. In models, such as TAEMS where only the functional dimension is specified, the organisation has nothing to "tell" to the agents when no plan or task can be performed. Otherwise, if only the structural dimension is specified as in AGR, the agents have to reason for a global plan every time the agents want to work collectively. Because the structure restricts the agent's options, even with a smaller search space of possible plans, the problem is deemed a difficult proposition. Moreover, because there is no organisational memory to store plans, the plans developed for a problem are lost. Therefore, in the context of open systems, we hypothesise that if the organisation model specifies both dimensions as in $\mathcal{M}$OISE$^{Inst}$ or TEAMCORE or a third one as in ISLANDER then the MAS that follows such a model can be more effective in leading the group behaviour to the desired objectives. On the agents' side, the models can develop richer reasoning abilities about agents and the organisation. Agents may gain more information on the possible cooperation (in terms of roles, groups, and on the possible goals, or on the performative structures) that may be conducted with other agents within the MAS.

Besides the aforementioned dimensions, the *deontic* and *normative* dimensions used in $\mathcal{M}$OISE$^{+}$ and ISLANDER or $\mathcal{M}$OISE$^{Inst}$ respectively address the agents autonomy problematic and consider organisations as normative constructs aiming at explicitly controlling the underlying MAS. While in other OMLs, the agents are supposed to be benevolent and must comply with the organisational specification (OS), the $\mathcal{M}$OISE$^{Inst}$ paradigm adds the possibility for agents to develop explicit reasoning on their autonomy with respect to the organisational constraints (Boissier and Gâteau, 2007).

### *2.2 Responsibility in MAS*

A number of works have focused on the responsibility held by agents in a MAS system. According to (Sommerville, 2007), such a responsibility can be considered as a duty, held by some agents, to achieve, maintain or avoid a given state, subject to conformance with organisational, social, and cultural norms. In this work, that responsibility is modelled using concepts as the: (a) rights necessary for the agent to achieve a task or obligation, (b) accountability, and (c) process of agent-task assignment that focus on the necessity to have an agent commitment before the assignment. Li et al. (Li and Hoang, 2009) have stressed the importance of using the responsibilities of a role in organisation to dynamically interact with the agents but does not go further to address the dynamic assignment of tasks to those agents. The authors proposed a role-interaction-organisation security model and applied the model to an e-health system, which is modelled as a MAS. The roles in the proposed model not only determine access rights passively, but also initiate requests to interact dynamically with the agents who meet the security requirements. That is to say that, the confidential the e-health data from unauthorised access is mandated. The interaction and the organisation models aid in identifying the actions and responsibilities that a role can assume in the system within the organisation and any dynamic interactions that the system can partakes. The authors in (Guemkam et al., 2011) have proposed an agent-based framework to support alert mechanism for power distribution systems by using a reputation based trust approach. The architecture provides a framework for dynamically assigning responsibilities to agents depending on the context of the crisis at hand. The aforementioned is done to manage the agent access rights towards critical information. The reputation-based trust is based on the similarity views between two agents during the assignment process. However, the scheme does not takes into account the assurance or body for evidence that is necessary for an agent to fulfil a task.

### *2.3 Monitoring of security*

MAS have also been extensively used for monitoring the security of a given system. For instance, the authors in (Boudaoud et al., 2000) proposed an intelligent multi-agent approach for the design of an Intrusion Detection System (IDS) with clear specification of the responsibilities of each agent in the monitoring. The scheme proposed operates at two layers. Firstly, at a higher level, the manager layer operates and manages the security of the network. Three different agent types operate at the manager layer. The *Security Policy Manager* agent manages the policies specified by the network security administrator. The *Intranet Manager* agents control the local agents that monitor the network traffic flow and report to them. These are managed by the *Extranet Manager* agent which assigns and delegates them intrusion detection tasks. Finally, the operations of the Extranet Manager agents are controlled by the policies of the Security Policy Manager agent.

Later in (Kolaczek and Juszczyszyn, 2007), Kolaczek et al. proposed an attack pattern ontology and a formal framework within a distributed multi-agent IDS. In this approach it is assumed that the network system consists of a set of nodes. Two types of agents are considered in the multi-agent system. *Monitoring agents* observe the nodes, process captured information and draw conclusions that are necessary to evaluate the current state of system security within their areas of responsibility. *Managing agents* are responsible for gathering information from *Monitoring agents* and generating reports about global threats and ongoing attacks. Each *Monitoring agents* monitors the corresponding area of responsibility that may consist of a set of network nodes.

Servin et al. introduced a reinforced learning-based approach for the design of an intelligent multi-agent IDS for detecting new and complex distributed attacks (Servin and Kudenko, 2008). In this reinforced learning architecture, each network sensor agent learns to interpret local state observations, and then communicates the information to a central agent higher up in the hierarchy. These central agents, in turn, learn to send signals up the hierarchy, based on the signals that they receive. The agent at the top of the hierarchy learns when to signal an intrusion alarm.

In (Abielmona et al., 2011), the challenges of applying MAS technology to the monitoring of territorial security before presenting their own architecture are discussed. The architecture which is referred to as retroactive cumulates both proactive and reactive features and enables a given agent to implement a reaction strategy based on the occurrence of an event in the environment. By insuring each agent records the strategy associated with a given event, the architecture allows an autonomous robot to learn over a period of time. This ultimately prepares these robots to respond in real-time to actual events in the environment, as they occur (Baig, 2012).

The BUGYO methodology (Ouedraogo et al., 2008) has adopted a MAS for the monitoring of the security assurance that is based on a hierarchy of three layers of agents. The first level of agents is a single agent that is embedded within the server. When the server receives a request to perform a security assurance evaluation, the server agent handles the request and identifies the appropriate multiplexer agent or MUX agent (using a role directory). Finally, probe agents trigger the associated probe in the event of receiving a measurement request from a MUX agent. The probe agents also collect measurements from instrumentations and transmit them to the MUX agents. Similarly, (Pham et al., 2008) present a security assurance evaluation approach based on attack graph. The MAS system proposed as implementation of the approach builds upon the BUGYO methodology. The concept of an attackability metric is introduced to characterise the possibility of attack along with other metrics for anomaly detection that assess both the static and dynamic visions of the underlying system.

In summary, the multi-agent based architectures for the monitoring of distributed systems examines the capacity of a software agent to achieve tasks through respect of norms or rules, and as mechanism to ensure security. However, the aforementioned models do not propose any method to reassign the responsibilities to peers when a given agent fails to fulfil its task.

## 3 The responsibility model

In this work, a responsibility is considered as a state assigned to an agent to denote the: (a) obligations concerning a task, (b) accountabilities regarding its obligations, and (c) rights and capabilities necessary to perform tasks (Guemkam et al., 2011).

### *3.1 Responsibility and role*

In our responsibility model, an obligation is a duty which links a responsibility with a task that must be performed. In $\mathcal{M}\text{OISE}^{Inst}$ a norms is an obligation or a permission that links a role with a mission to achieve.

More precisely, the accountability concept denotes a duty to justify achievement, maintenance or avoidance of some given state to an authority under threat of sanction (Stahl, 2006). Therefore, the accountability parameter contributes to the valuation of trust. An agent's right encompasses facilities required by an agent to fulfil the desired obligations, such as the access right that the agent gets once it is assigned with a responsibility. Capability describes the required qualities, skills or resources necessary to perform a task. The capability parameter depends on number of parameters relating the agents. These include the: (a) ability to make decisions, (b) processing time, and (c) ability to analyse a problem and the location within the network. The commitment pledged by the agent represents the engagement to fulfil a task. The commitment concept has been the subject of many researches in MAS as explained in great detain in (Singh, 2008).

$\mathcal{M}\text{OISE}^{Inst}$is used to specify an organisation with the help of four dimensions (Gâteau et al., 2005): (a) structural specification (SS), (b) functional specification (FS), (c) contextual specification (CS), and (d) normative specification (NS). The *structural specification* defines the MAS structure with the notions of *roles*, *groups* and *links*. In the *functional specification*, goals that are to be achieved by the organisation are structured into *missions*. To deal with applications in evolving environment, a *contextual specification* (CS) captures design-time *a priori* constraints on the evolution of the organisation as a set of contexts and transitions between the specifications.

The CS is not used in this work. The *normative specification* (NS) glues all of the specifications (SS, FS and CS) in a coherent and normative organisation with the help of norms. In $\mathcal{M}$OISE$^{Inst}$, norms define rights (i.e. *permission*) and duties (i.e. *obligation*, *prohibition*) for agents while playing a role to execute a *mission* in a particular *context*.

A parallel is performed between the responsibility model and the normative organisation model of MAS by matching role with responsibility, mission with task, the set of goals to achieved to accomplish the mission with the capabilities required to perform the task and right, and obligation with norm (which could be a permission or an obligation). The detailed responsibility model matched to a simplified view of $\mathcal{M}$OISE$^{Inst}$ is illustrated in Figure 1.

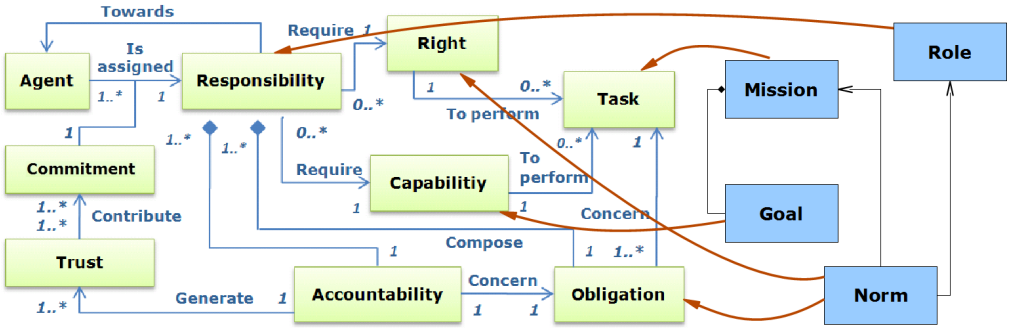


**Figure 1** Match between responsibility model and normative organisation model.

### *3.2 Assurance and trust*

Although a plethora of conditions may need to be fulfilled for expecting an agent to meet its responsibilities, it is imperative that the following ones are met:

1. Rights: the set of rights entrusted to the agent must be such that they enable satisfaction of the agent's obligations.
2. Capability: the overall capability assigned to an agent must be below the agent's intrinsic capability. Moreover, such capability should enable the agent to fulfil its obligations.
3. Level of trust: should be higher or equal to a minimum (predefined) threshold.

Based on the above requirements the assurance for an agent fulfilling the obligation should be based on: *Assurance for fulfilment of obligation "o" by an agent with right "R", capability "C", and trust "$T_{min}$": $A_o(R, C, T)$.*

No assurance:

$$A_o(R, C, T_{min}) = 0 if (R_o \notin R) \cup (C_o \notin C) \cup (T_p \geq T_{min}) \quad (1)$$

Otherwise:

$$A_o(R, C, T) = 1, \quad (2)$$

where

$R$: the current rights of the agent;
$C$: the current capabilities of the agent;
$R_o$: the set of rights necessary for fulfilling obligation $o$;
$C_o$: the set of capabilities necessary for fulfilling obligation $o$;
$T_p$: the trust at period $p$.

Relations (1) and (2) imply that the satisfaction of an obligation can only be guaranteed if the set of rights allocated to the agent and the current capabilities are both subsets of the set of rights and capabilities required for the satisfaction of the obligations and if the trust level at

period ($T_p$) is higher or at least equal to the reference $T_{min}$. It is noteworthy to mention that more than one agent may fulfil such requirements and subsequently, the decision to select one of those as an alternative to a faulty agent will be based on the highest level of $T_p$.

From the MAS point of view the main concern is how to develop an organisation infrastructure that ensures the satisfaction of the organisational constraints and norms (e.g. agents playing the right roles, committing to the allowed missions). Many implementations of the organisation infrastructure follow the general architecture depicted in Figure 2. Domain agents are responsible to achieve organisational goals and use an *organisational proxy* component to interact with the organisation. The *organisational layer* is responsible to bind all agents in a coherent system and provides some services for them (Kitio et al., 2007). In particular, they are in charge of verifying above conditions in order to decide of a potential reorganisation. The level of trust $T_p$ of each of the component is provided by the organisational layer based on direct information as detailed in (Guemkam et al., 2011).

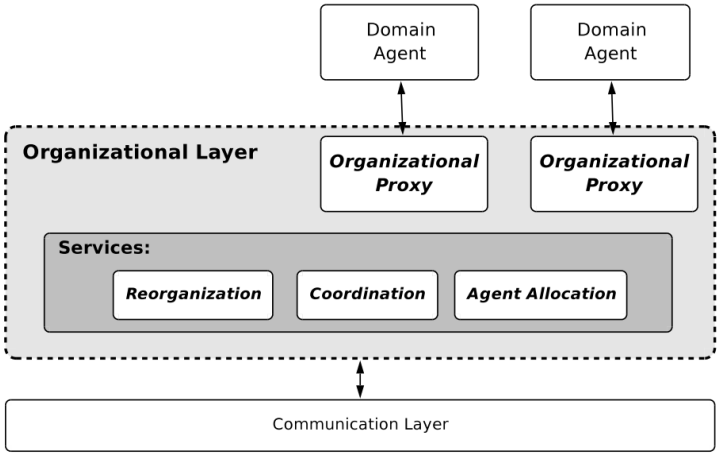


**Figure 2** Common organisation implementation architecture for open MAS (Kitio et al., 2007).

The next section describes a scenario that we adopt to illustrate the reassignment of an obligation based on assurance and trust values.

## 4 The broadcasting mechanism scenario

The broadcasting mechanism (as depicted in Figure 3) aims at sending alerts to the population using media, such as Short Message Service (SMS) whenever a severe weather alert occurs. For that, sensors are disseminated on three layers corresponding to geographical areas (city, region, or country) and the sensors retrieve information pertaining to pressure, temperature, and electric voltage from probes located within a weather station and from the electrical grid. Regarding the different layers, sensors and aggregators have specific responsibilities:

- The Alert Correlation Engine (ACE) collects, aggregates, and analyses weather information from the probes deployed over the network and weather stations.
- Confirmed alerts are sent to the Policy Instantiation Engine (PIE). The PIE receives confirmed alert from the ACE, sets the severity level, and the extent of the geographical response. The PIE also instantiates high level alert messages to be deployed.
- Finally, the high level alert messages are transferred to the Message Supervising Point (MSP).

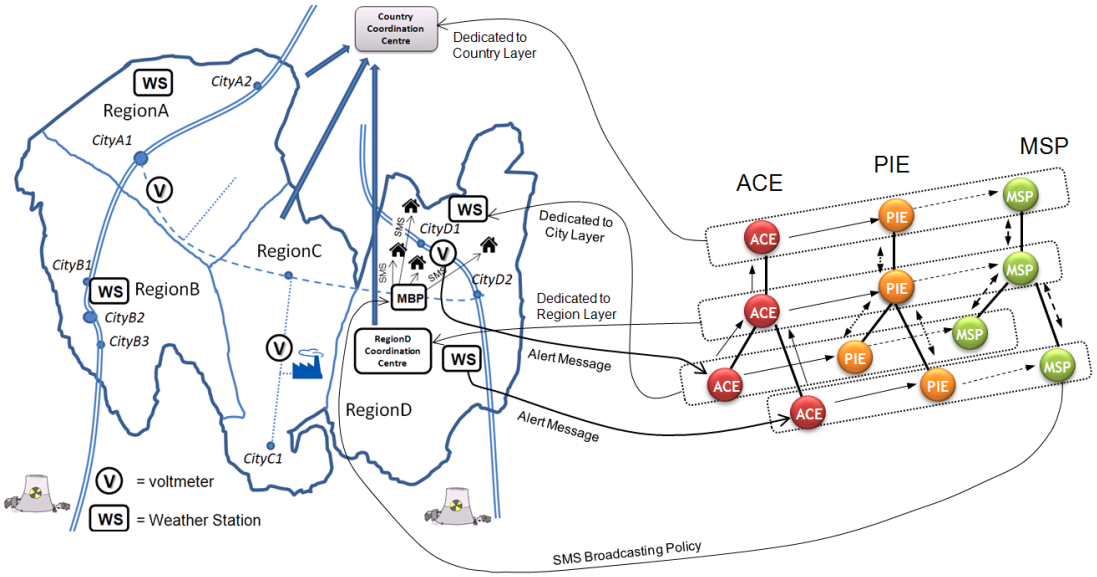


**Figure 3** Broadcasting mechanism inside.

## 5 Assigning the responsibilities to the agents

### *5.1 Implementation*

The architecture is composed of different types of agents that play a collaborative role. The agent architecture presented in Figure 3 is based on the Reaction after Detection (ReD) project (Gâteau et al., 2009). It proposed a solution to enhance the detection/reaction process and to improve the overall resilience of critical infrastructures. We extended the aforementioned architecture to accommodate the need for the responsibilities reassignment. The main agents involved include the: (a) Alert Correlation Engine (ACE), (b) Policy Instantiation Engine (PIE), (c) Message Supervising Point (MSP), and (d) Message Broadcasting Point (MBP).

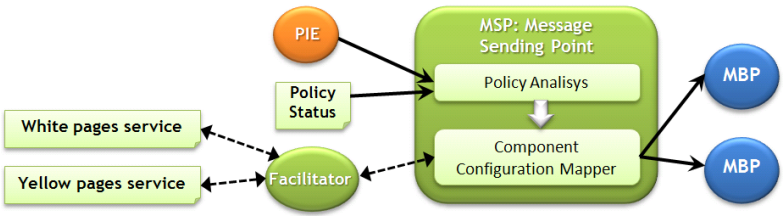


**Figure 4** Message Supervising Point (MSP) architecture.

The MSP (Figure 4) is composed of two modules: (a) the Policy analysis (PA) and (b) the Component Configuration Mapper. The PA is in charge of analysing the policies previously instantiated by the PIE. For that, the Policy Status database stores all of the communication policies and the corresponding status (in progress, not applicable, bypassed, enforced, removed) so that the PA module can check the consistency of the newly received message to be deployed. The Component Configuration Mapper module selects the appropriate communication channel.

The MBP is in charge of receiving the generic alert messages from the MSP. Then a specific parser converts the incoming alert message to the appropriate format according to the channel. Figure 5 presents two different kinds of MBPs. Different communication channels (e.g. SMS, e-mail, micro-blogging) to send alerts to citizens, hospitals, etc. are used. Consequently, our electric blackout prevention system is easily extensible for future communications facilities.

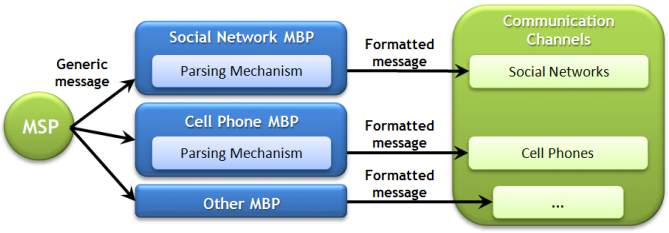


**Figure 5** Message Broadcasting Point (MBP) architecture.

To consider the centralised trust of each agent, the organisational layer, specified with $\mathcal{S}$YNAI and instantiated here with $\mathcal{U}$TOPIA from (Schmitt et al., 2011), maintains a database of levels of trust of agents regarding the status of the accomplishment of their missions. This means that the MBP, the ACE, and the MSP have dedicated levels of trust. $\mathcal{U}$TOPIA's context permits to assign and manage the responsibilities of the components, such as MBP and MSP. Thereby, $\mathcal{U}$TOPIA controls the assignment of roles in an intelligent manner by taking into account the current level of assurance of a given norm to be respected as well as the trust level $T_p$ of the agents.

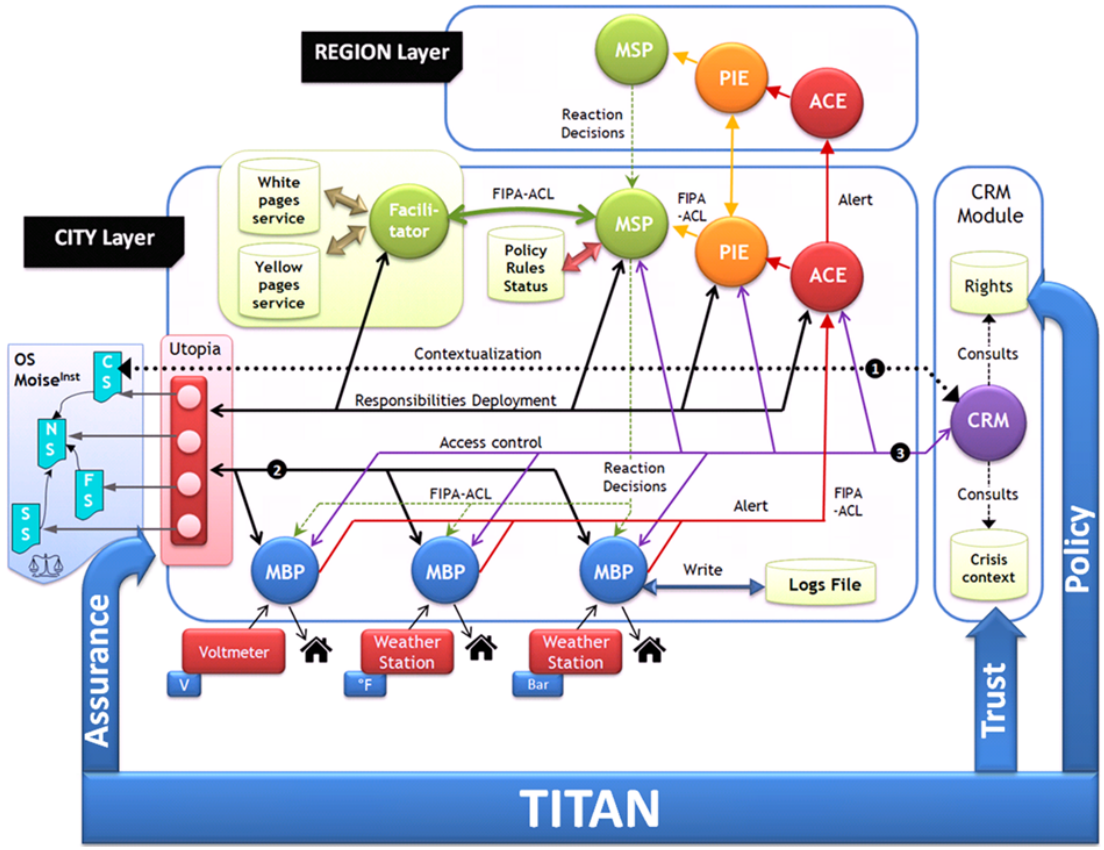


**Figure 6** Detailed reaction architecture for power distribution adaptation based on weather parameters.

Figure 6 introduces the extended ReD architecture illustrated with the weather broadcast alert system. The flow begins with an alert detected by a probe. The alert is sent to the ACE agent (City Layer) that confirms or not the alert to the PIE. Afterwards, the PIE decides to apply new policies or to forward the alert to an ACE from a higher layer (Region Layer). The PIE agent sends the policies to the MSP agent that decides which MBP is able to transform the high level

| Capabilities & Rights | Agent's Obligations | Mapping of Capabilities to Obligations | Mapping of Rights to Obligations |
|---|---|---|---|
| **Capabilities**<br>$C_1$: Is on the same network as the component to control<br>$C_2$: Be able to communicate with the MSP<br>$C_3$: Be able to communicate with the facilitator agent<br>$C_4$: Have enough computing resource to monitor the component to control<br>$C_5$: Be able to communicate with the MAS Management layer<br>$C_6$ : Must be able to encrypt data<br>$C_7$ : Be able to communicate securely with the ACE<br>**Rights**<br>$R_1$ : Allow to read log file on the concerned network component<br>$R_2$ : Allow to write log in the central logs database<br>$R_3$ : Be able to read the Policy in the MAS management layer<br>$R_4$ : Allow to read and right in the alert database | $O_1$: Must retrieve the logs from the component it monitors | $C_1$, $C_4$, $C_6$, $C_7$ | $R_1$, $R_2$, $R_4$ |
| | $O_2$: Must provide an immediate reaction if necessary | $C_1$, $C_2$, $C_3$ | $R_1$, $R_2$, $R_3$ |
| | $O_3$: Must communicate with the facilitator in order to get the address of the other components (MSP, ACE) | $C_3$ | |
| | $O_4$: Must report the incident to the ACE in a secure way | $C_5$, $C_6$, $C_7$ | $R_5$ |

**Table 1** Message Broadcasting Point (MBP) Responsibilities Specifications.

alert message into an understandable format for the selected communication channel (Feltus, 2010). To manage access rights, we incorporate to ReD a Context Rights Management module (CRM) (see Figure 6). The CRM is in charge of providing and amending the access rights to agents. For that, the CRM is linked to the database storing information on the agents' rights. It is also linked to the Context Manager of $\mathcal{U}$TOPIA to detect changes in the context, particularly when an agent becomes short.

### *5.2 Organisation*

Based on the agents' responsibility model shown in Figure 1, we define the responsibilities of each agent within the architecture. Table 1 summarises the necessary capabilities and the rights for the MBP to accomplish a given obligation. We observe that the responsibilities include obligations, such as the obligation $O_1$ to retrieve the logs from the component it monitors and $O_2$ to provide an immediate reaction if necessary. To perform the latter obligation $O_2$, it must have the capabilities to be on the same network as the component it controls, such as voltmeter, thermometer or barometer ($C_1$), to be able to communicate with the MSP ($C_2$), with the facilitator agent ($C_3$), etc. It also must have the rights $R_1$ to read the log file on the concerned network component, $R_2$ to write the log in a central logs database and finally $R_3$ to be able to read the Policy in the MAS management layer.

### *5.3 Re-organisation*

In the considered scenario, we assume that the occurrence of the adverse event results in some changes in the rights and capabilities of the agents to fulfil their respective obligations. Table 2 provides the new capabilities and rights of the agent as well as the corresponding assurance values to meet a given obligation. Such assurance value is based on the metrics provided in Section 3. After taking into account the specifications of the responsibilities associated with each agent provided in Table 1, one can assess whether current rights, capabilities, and trust level of a MBP agent can be sufficient to fulfil a given obligation. Let us consider for instance some of the information in Table 2. The actual status of MBP is such that it will not be able to fulfil the respective obligation $O_2$. The obligation to provide an immediate reaction is hindered by the

| **Obligations Concerning Tasks** | **Trust level** | **Current agents' capabilities** | **Current agents' obligations** | **Assurance of obligation fulfilment** |
|---|---|---|---|---|
| $O_1$: Must retrieve the logs from the component it monitors | $T_p$= 0.5 | $C_1$, $C_4$,$C_6$, $C_7$ | $R_1$, $R_2$, $R_4$ | 1 |
| $O_2$: Must provide an immediate reaction if necessary | | $C_1$, $C_3$ | $R_3$ | 0 |
| $O_3$: Must communicate with the facilitator tin order to get the address of the other components (MSP, ACE) | | $C_3$ | | 1 |
| $O_4$: Must report the incident to the ACE in a secure way | | $C_5$, $C_6$, $C_7$ | $R_5$ | 1 |

**Table 2** Rights and capabilities of Message Broadcasting Point (MBP) at time t.

fact that the MBP lacks the capability to communicate with the MSP ($C_2$). This means that any appropriate policy cannot be grounded to the MBP and be implemented in case of abnormally within the infrastructure.

In the event that the assurance value for meeting a given obligation is "0", as in the cases discussed above, such an obligation is devolved to an agent having the required rights and capabilities and belonging to the same group. For instance, in our scenario, when the GSM/GPRS network is down, the MBP-SMS agent is replaced by the MBP-EMAIL agent. The decision taken by $\mathcal{U}$TOPIA is to re-organise the agents by assigning this role to another agent having rights and capabilities to accomplish the missions and having a trust level higher than the $T_p$ needed.

## 6 Conclusion

This paper presented a novel approach to address the dynamic reassignment of an agent's responsibilities to its peer when it becomes unable to carry out its obligations. Indeed, MAS are widely used for the monitoring of distributed systems, but the assignment of the agents' functions is undertaken prior to the MAS system deployment. The proposed model exploits the concepts of assurance and trust as the indicators for identifying respectively, when an agent becomes unable to meet its obligation, and for selecting the alternative peer that is believed to have a higher reliability for carrying out the task. An instantiation of the model has been presented on a weather station use case scenario. The architecture is developed using ReD and UTOPIA and has been shown to be effective during our first simulations, in ensuring the continuity of the monitoring in the event an agent was to lose some of its capabilities or rights. We simulated several responsibility assignment and reassignment for validating the approach and monitored the behaviours of the MBP, MSP and ACE deployed in different location with different parameters. However, the instances discussed in this paper only considered one faulty agent at a time. Therefore, our future work will be directed towards the consideration of more complex scenarios, such as agents failing simultaneously and consecutively after the occurrence of an adverse event.

## Acknowledgement

This work was partially funded by TITAN Project (C08/IS/21), financed by the National Research Fund of Luxembourg.

## References

Abielmona, R. S., E. Petriu, M. Harb, and S. Wesolkowski (2011). Mission-driven robotic intelligent sensor agents for territorial security. *IEEE Computational Intelligence Magazine 6*(1), 55–67.

Baig, Z. A. (2012). Multi-agent systems for protecting critical infrastructures: A survey. *Journal of Network and Computer Applications 35*(3), 1151—1161.

Boissier, O. and B. Gâteau (2007). Normative multi-agent organisations: Modeling, support and control. In *Proc. Dagstuhl Seminar 07122*.

Boudaoud, K., H. Labiod, R. Boutaba, and Z. Guessoum (2000). Network security management with intelligent agents. In *Proc. Network Operations and Management Symposium (NOMS 2000)*, pp. 579–592. IEEE/IFIP.

Coutinho, L., J. Sichman, and O. Boissier (2007). Organisational modeling dimensions for multi-agent systems. In *Proc. Workshop Agent Organizations: Models and Simulations*.

Demazeau, Y. (1995). From interactions to collective behaviour in agent-based systems. In *Proc. 1st. European Conference on Cognitive Science*, pp. 117–132.

Esteva, M., J. Padget, and C. Sierra (2002). Formalizing a language for institutions and norms. In *Proc. 8th International Workshop on Intelligent Agents VIII*, Volume 2333 of *LNAI*, pp. 348–366. Springer.

Feltus, C. (2010). A security decision-reaction architecture for heterogeneous distributed network. In *Proc. International Conference on Availability, Reliability, and Security (ARES 2010)*, pp. 1–8.

Ferber, J. and O. Gutknecht (1998). A meta-model for the analysis and design of organizations in multi-agent systems. In *Proc. Third International Conference on Multi-Agent Systems (ICMAS 1998)*, pp. 128–135. IEEE.

Ferber, J., F. Michel, and J. Baez (2004). AGRE: Integrating environments with organisations. In *Environments for Multi-Agent Systems*, Volume 3374 of *LNCS*, pp. 48–56. Springer.

Gâteau, B., O. Boissier, D. Khadraoui, and E. Dubois (2005). $\mathcal{M}$OISE$^{Inst}$: An organizational model for specifying rights and duties of autonomous agents. In *Proc. 1st International Workshop on Coordination and Organisation (CoOrg 2005)*.

Gâteau, B., D. Khadraoui, and C. Feltus (2009). Multi-agents system service based multi-agents system service based platform in telecommunication security incident reaction. In *Proc. Global Information Infrastructure Symposium (GIIS 2009)*, pp. 1–6. IEEE.

Guemkam, G., C. Feltus, P. Schmitt, C. Bonhomme, D. Khadraoui, and Z. Guessoum (2011). Reputation based dynamic responsibility to agent assignment for critical infrastructure. In *Proc. 2011 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, Volume 2, pp. 272–275. IEEE.

Hübner, J., J. Sichman, and O. Boissier (2002). A model for the structural, functional, and deontic specification of organizations in multiagent systems. In *Proc. 16th Brazilian Symposium on Artificial Intelligence (SBIA 2002)*, Volume 2507 of *LNAI*, pp. 118–128. Springer.

Jennings, N. R. (1999). Agent-oriented software engineering. In *Proc. 9th European Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW 1999)*, Volume 1647 of *LNCS*, pp. 1–7. Springer.

Kitio, R., O. Boissier, J. F. Hübner, and A. Ricci (2007). Organisational artifacts and agents for open multi-agent organisations: "giving the power back to the agents". In *Proc. 2007 international conference on Coordination, organizations, institutions, and norms in agent systems III (Coin 2007)*, pp. 171–186.

Kolaczek, G. and K. Juszczyszyn (2007). Traffic and attack pattern analysis for multiagent distributed intrusion detection system. In *Proc. Intelligent Systems and Knowledge Engineering (ISKE 2007)*, pp. 733–739. Atlantis Press.

Lesser, V., K. Decker, W. T., N. Carver, A. Garvey, B. Horling, D. Neiman, R. Podorozhny, M. NagendraPrasad, A. Raja, R. Vincent, P. Xuan, and X. Zhang (2004). Evolution of the GPGP/TAEMS domain-independent coordination framework. *Autonomous Agents and Multi-Agent Systems 9*(1), 87–143.

Li, W. and D. Hoang (2009). A new security scheme for e-health system. In *Proc. International Symposium on Collaborative Technologies and Systems*, pp. 361–366.

Ouedraogo, M., D. Khadraoui, B. De Remont, E. Dubois, and H. Mouratidis (2008). Deployment of a security assurance monitoring framework for telecommunication service infrastructure on a voip system. In *Proc. New Technologies, Mobility and Security Conference (NTMS 2008)*.

Pham, N., L. Baud, P. Bellot, and M. Riguidel (2008). A near real-time system for security assurance assessment. In *Proc. 3rd International Conference on Internet monitoring and protection*, pp. 152–160. IEEE.

Pynadath, D. and M. Tambe (2003). An automated teamwork infrastructure for heterogeneous software agents and humans. *Autonomous Agents and Multi-Agent Systems 7*(1–2), 71–100.

Schmitt, P., C. Bonhomme, J. Aubert, and B. Gâteau (2011). Programming electronic institutions with utopia. *Information Systems Evolution 72*, 122–135.

Servin, A. and D. Kudenko (2008). Multi-agent reinforcement learning for intrusion detection. In *Adaptive Agents and Multi-Agent Systems III*, Volume 4865 of *LNCS*, pp. 211–223. Springer.

Singh, P. (2008). Semantical considerations on dialectical and practical commitments. In *Proc. 23rd national conference on Artificial intelligence (AAAI 2008)*.

Sommerville, I. (2007). *Responsibility and Dependable Systems*, Chapter Models for responsibility assignment, pp. 165–186. Number 8. Springer.

Stahl, B. (2006). Accountability and reflective responsibility in information systems. In *The Information Society: Emerging Landscapes*, pp. 51–68. Springer.

Wooldridge, M. (2002). *An Introduction to Multi-Agent Systems*. John Wiley & Sons.